

## **SDOE 650: System Architecture and Design**

Homework Project

Colin George

Due Date: June 14, 2014

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

1. What need does the system address - that is, what purpose does it serve?  
Also, provide some background details on why the need exists.

**Background:**

*Our cables have one basic requirement that invoke hundreds of other requirements for each cable. The basic requirement is that the cable shall perform the functions defined in their respective Compatibility Definition (CD) drawing with reliability. The CD defines the functionality of the cables for a given life of the cable in the expected Environments. Rather than describing the full range of requirements that the component needs to survive, the CD invokes a spec that is specific to the weapons system that the cable is being qualified for. The spec that the CD invokes is called an Environmental Specification (ES). Our entire set of requirements flow out of three documents the CD, the ES, and an mechanical envelope (ME) which describes the physical space it must be confined to in order for the cable to fit in the system.*

*The proposed system is a test system that verifies the cables functionality in the expected environments defined in the ES. Verification methods include test, inspect, demonstrate, and analyze. Since we are defining the architecture for a test system we will focus on the customer expectations and requirements that will be satisfied or verified via testing.*

**System objective:**

*This cable test system will verify all requirements that the PRT miens to verify through testing. This system will also consider customer expectation in developing test flow, test levels, cost and schedule. The output from this testing will feed to other efforts. First and foremost the output of this testing will be used in the qualification of cables in the. Secondly, this data package will be used to do a margin assessment. The customer wants a specific flavor of margin assessment. The costumer wants all of the assessments to be consistent so that they could be integrate and rolled up to increase the fidelity of the systems margin assessment.*

2. Provide a context diagram (external systems diagram) in CORE that shows your

system and its external interfaces to the active stakeholders.

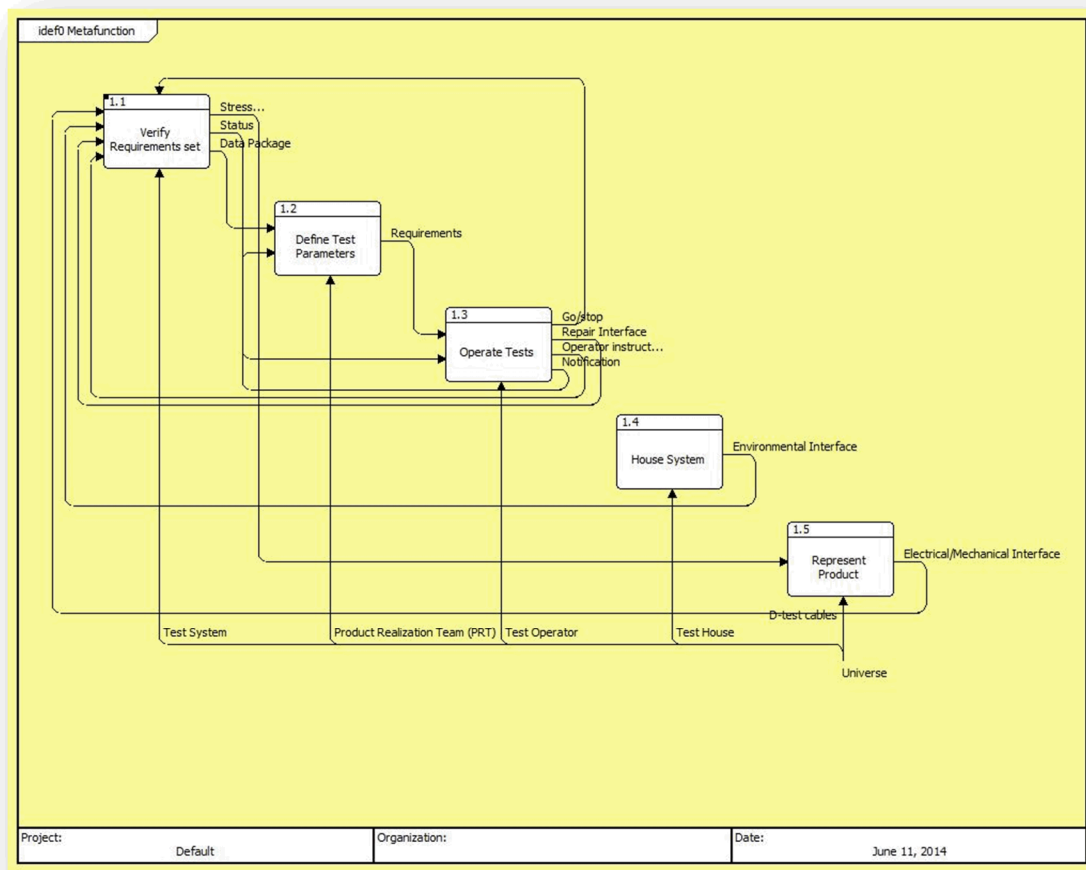
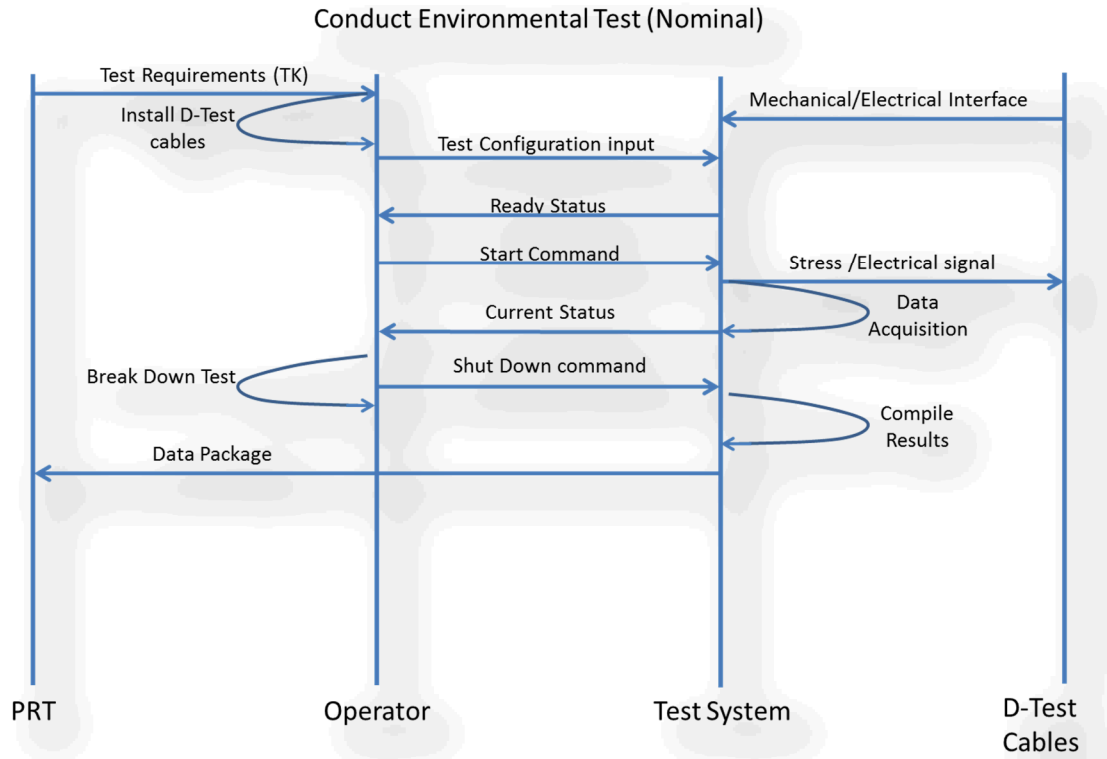
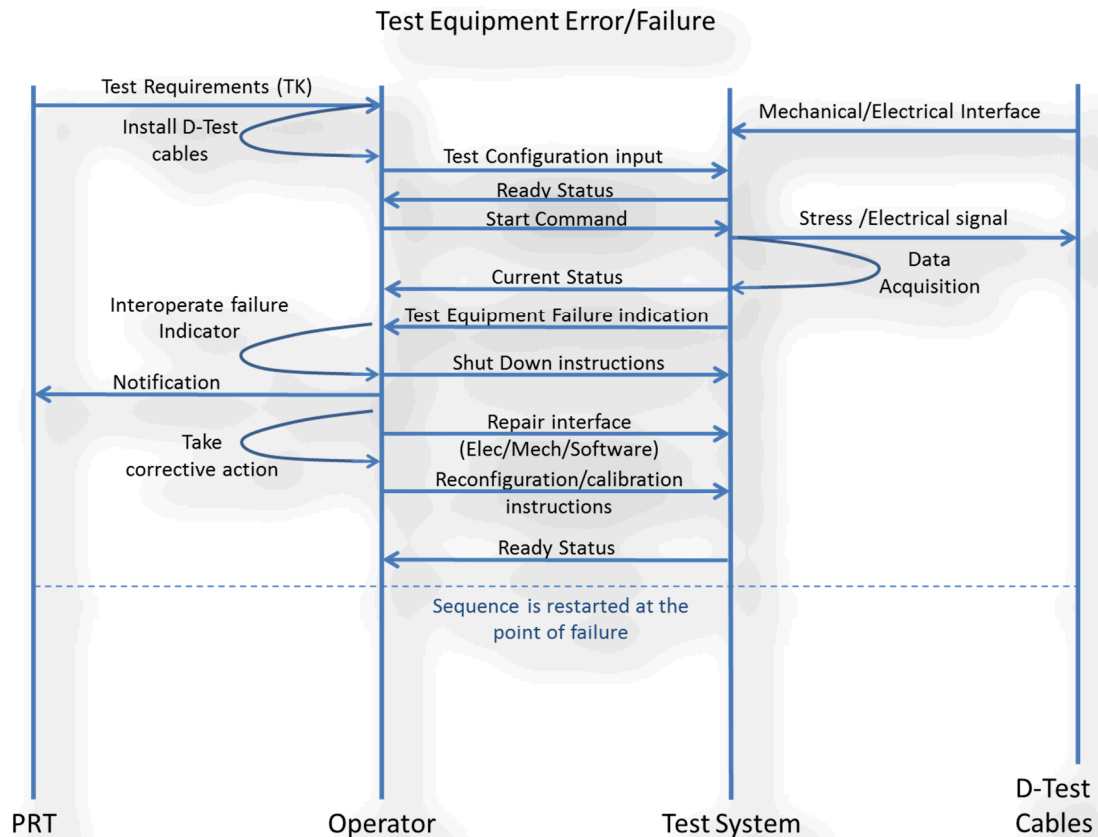


Figure 1 Test System Context Diagram

3. Develop sequence diagrams that illustrate the interactions for four architecturally-significant' usage scenarios.



**Figure 2 Conduct Environmental Test Sequence Diagram**



**Figure 3 Test Equipment Error/Failure Sequence Diagram**

### Operator Error

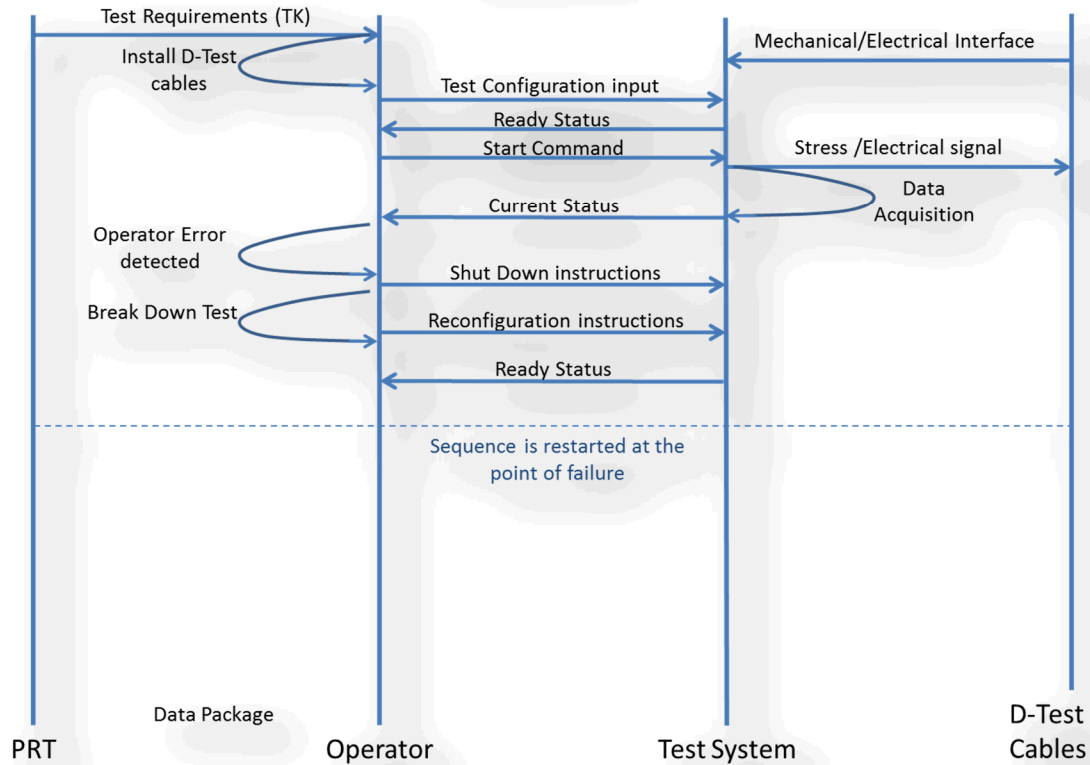


Figure 4 Operator Error Sequence Diagram

### Cable Failure

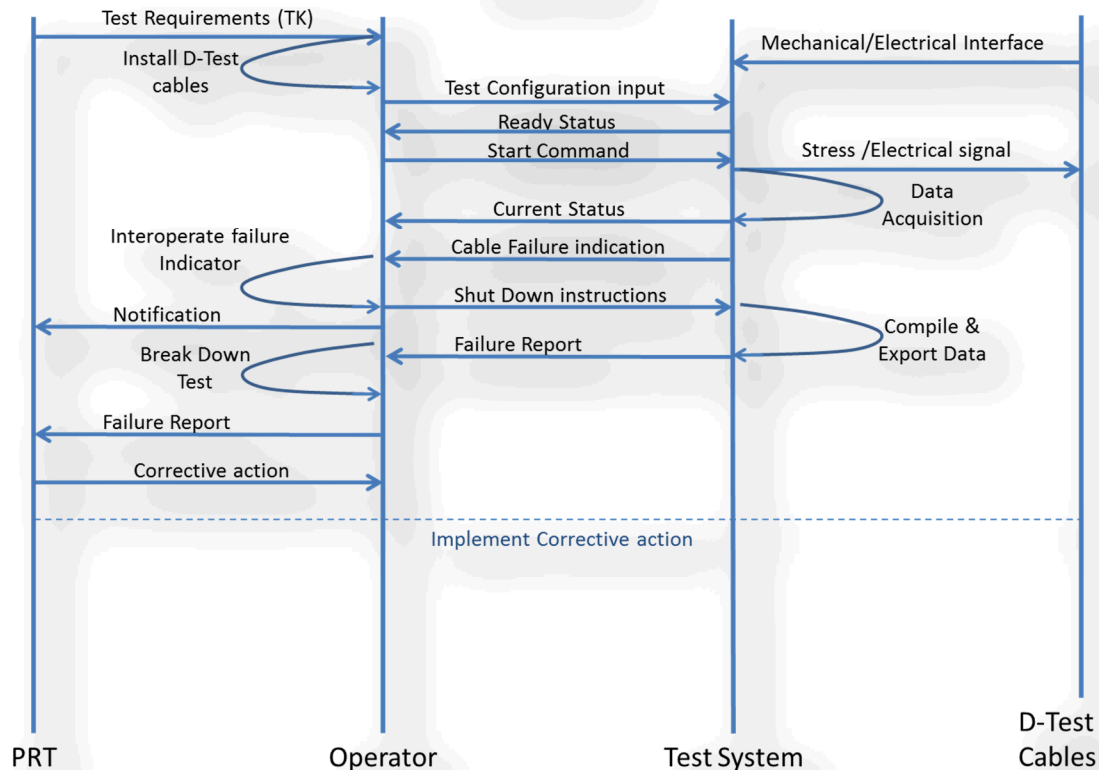
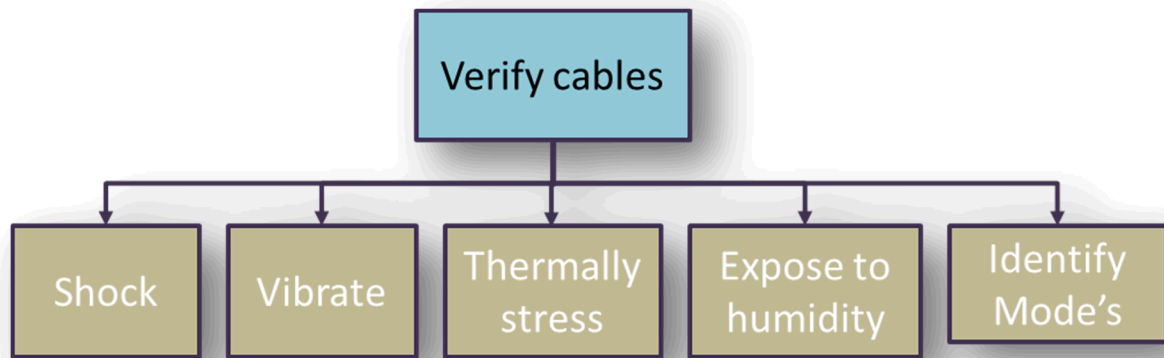


Figure 5 Cable Failure Sequence Diagram

4. Decompose the system function to create a first-level functional architecture. Provide a minimum of five functions in the first-level architecture.

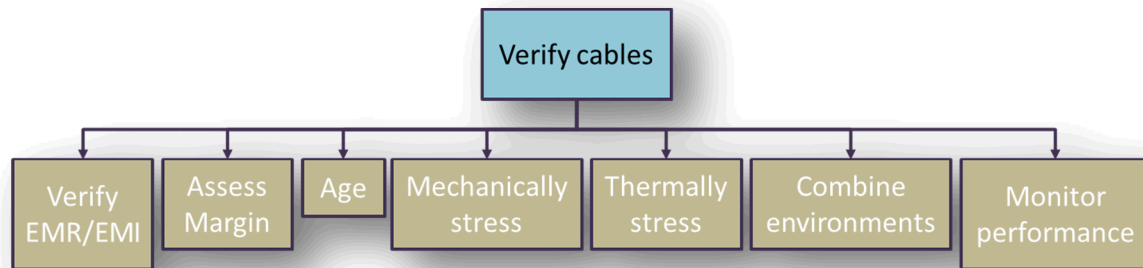


**Figure 6 Function Architecture Following Physical Architecture**

*In considering Physical Architecture, Functional Requirements, and Operating Modes as the methods for defining the functional partitions, I have chosen the method of following the Physical Architecture approach. Figure 6 is the functional architecture that is based on the physical architecture. Although it was difficult to decide between the Physical Architecture and the Functional Requirements partition, I settled on the above architecture because of the following reasons:*

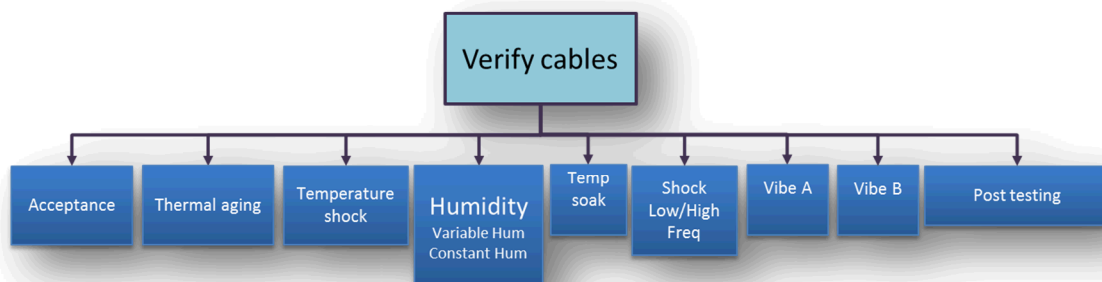
1. *The components must be exchangeable: It turns out that we have already issued a contract to accomplish Environmental testing. This means that there is a suite of equipment that we will be using and there is a limit to our equipment options. In addition the contracted test house has the latitude to use more than one model or type of equipment as long as it can meet a minimum spec. Considering these constraints it seems that choosing an architecture that allows component interchangeability is critical.*
2. *Our architecture must be developed with a limited catalog of components: It is known that there is a limited number of test houses and equipment that we can use and can meet our requirements. If I used another functional partition it would be more likely that I would develop an architecture that is not possible given our resources. I will create the architecture based on the list of components that we have at our disposal and negotiate the customer requirements that we can not meet.*
  - a. Document how else you could have partitioned the system function. What are the advantages and disadvantages of partitioning it the way

you did? I am looking for alternative functional partitioning(s) that make sense, and thoughtful rationale of why you picked the one that you did as opposed to the alternative(s). Provide at least a one page discussion of your rationale.



**Figure 7 Function Architecture using Functional Requirements**

*The functional Architecture using a partition based on Functional Requirements was another good option after the Physical Architecture approach. There are a significant number of Environmental requirements and customer expectation that will need to be verified with the proposed system. It would be beneficial to see the mapping between the requirements and the system. In fact it would seem that you would be more likely to verify all of the requirements if the architecture had been designed in light of the requirements. Since this systems function is to “verify” the cable requirements, the Functional Requirements approach is a very good consideration. Unfortunately, I believe the architecture would become confusing and complicated after the first level. If we chose this partition we would gain some one-to-one mapping but we would have a system that would suffer in the end because of it.*



**Figure 8 Function Architecture using Operating Modes**

*An Architecture based on the operating modes of the system was the last choice. This option seemed achievable but did not have any clear benefits. Furthermore*

*the operating modes were based on our current plan but we have the latitude to define the operating modes that we want. It seems better to create an architecture in light of your most challenging constraints than to do so in light of an elements of the system that is not a constraint.*

5. Develop an IDEF0 model in CORE for the system function and the functional decomposition. Assign the external inputs and outputs to the system function and to the appropriate sub-functions. Create the necessary internal inputs and outputs to the first level architecture.

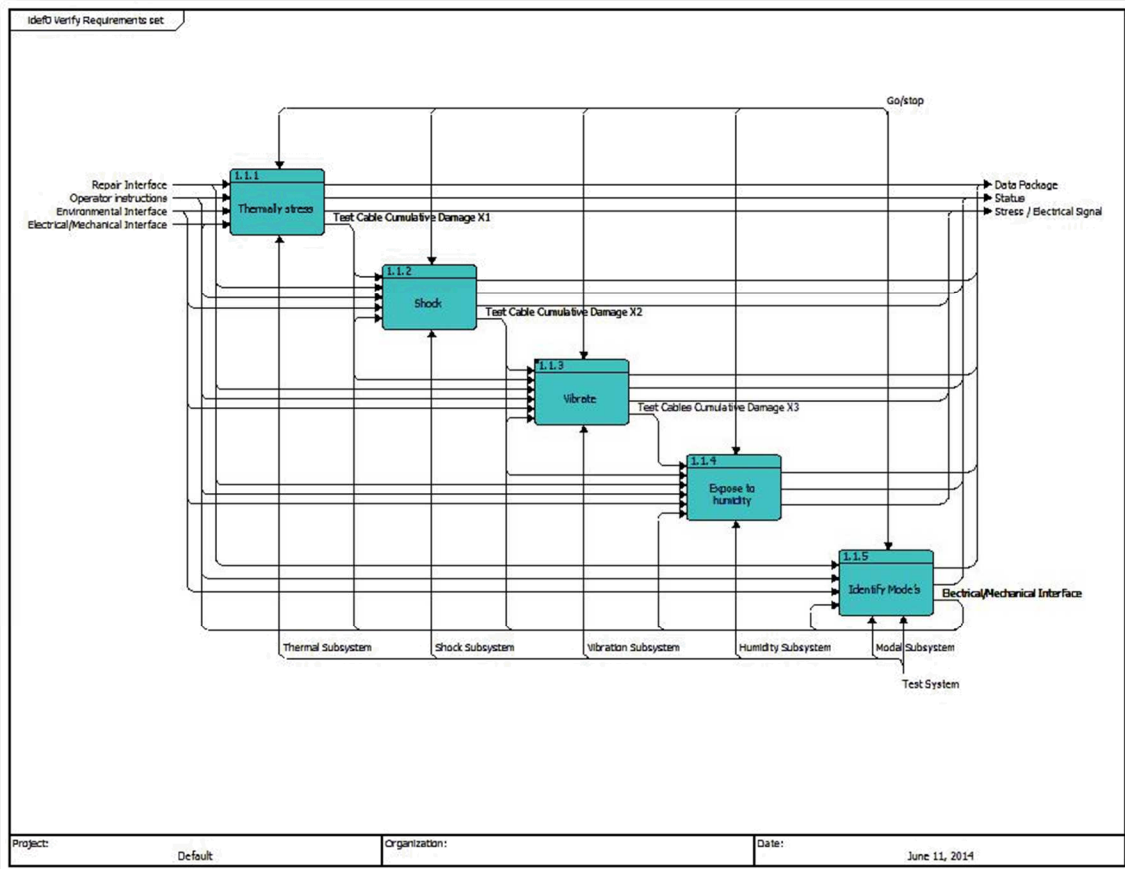


Figure 9 Verify Requirements set IDEF0

- a. Document a trace of your four scenarios through your architecture.

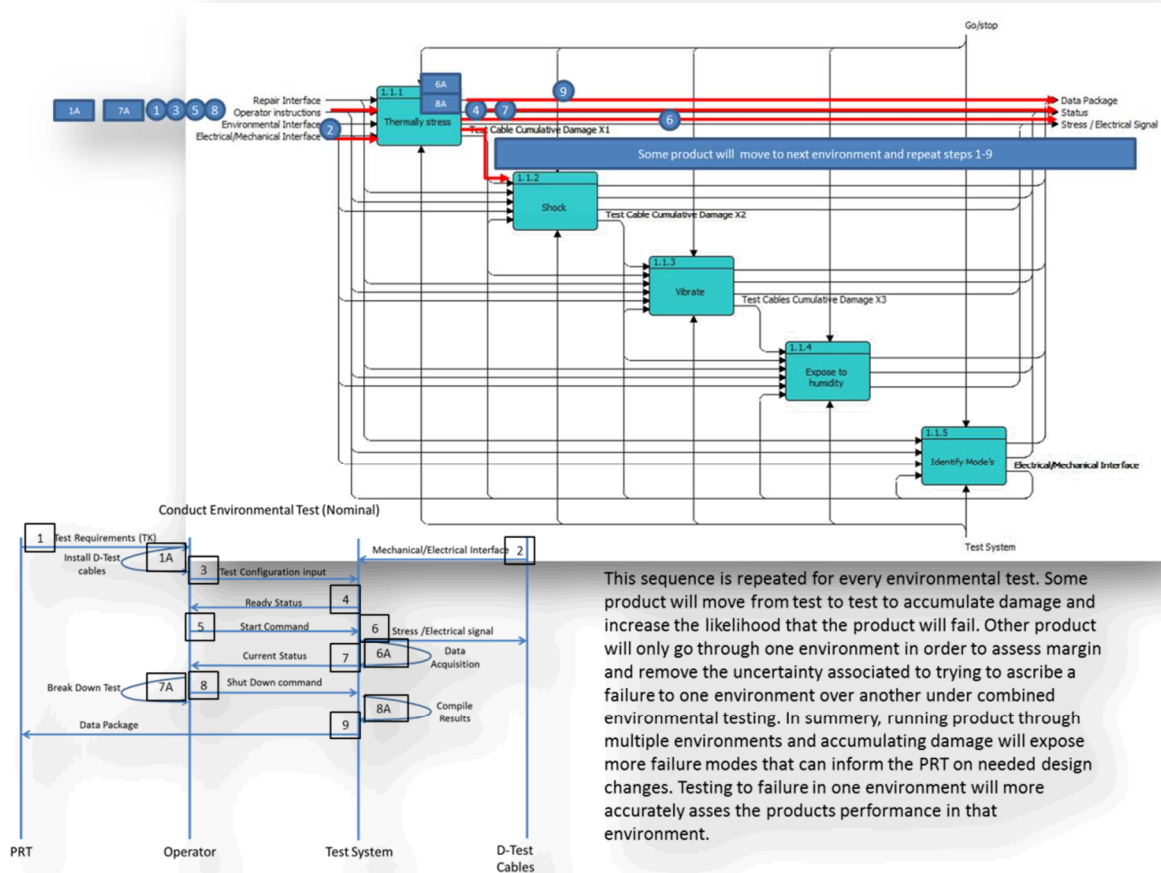


Figure 10 Conduct Environmental Test Trace

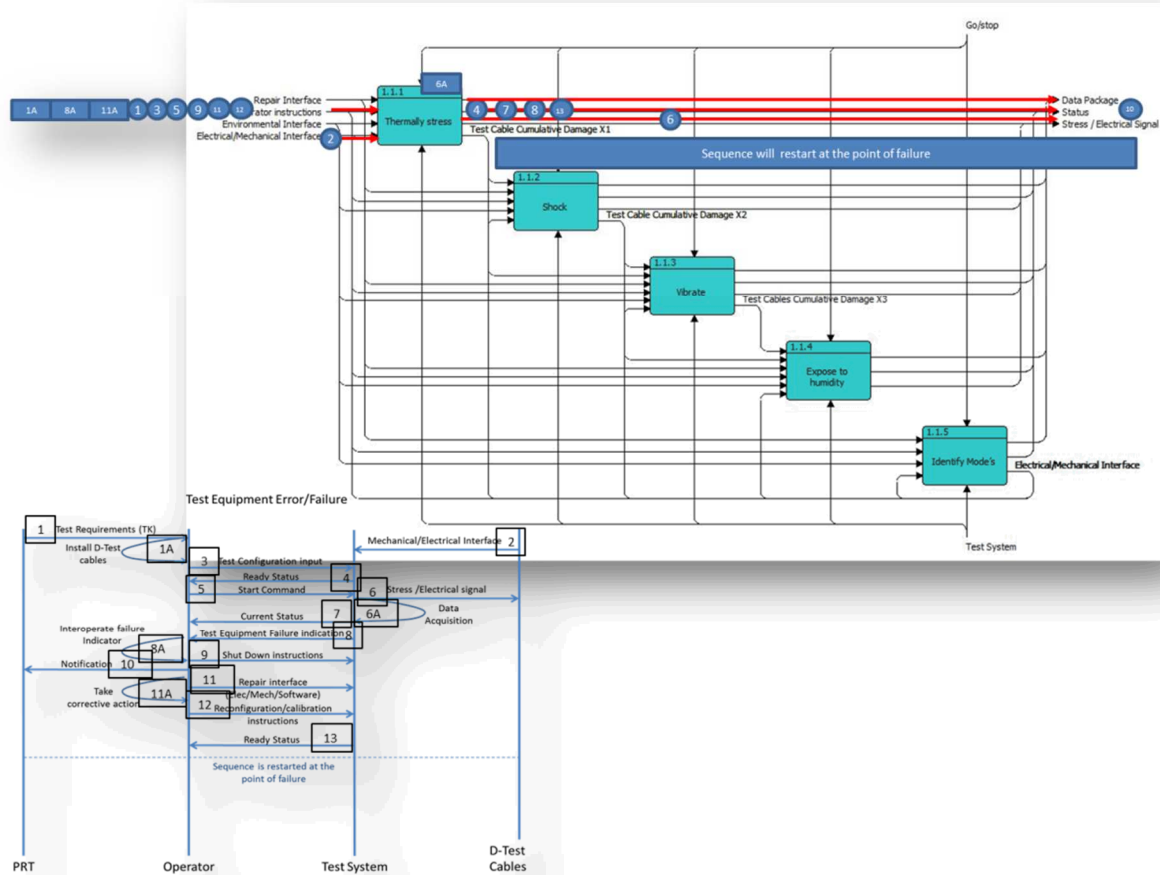


Figure 11 Test Equipment Error/Failure trace

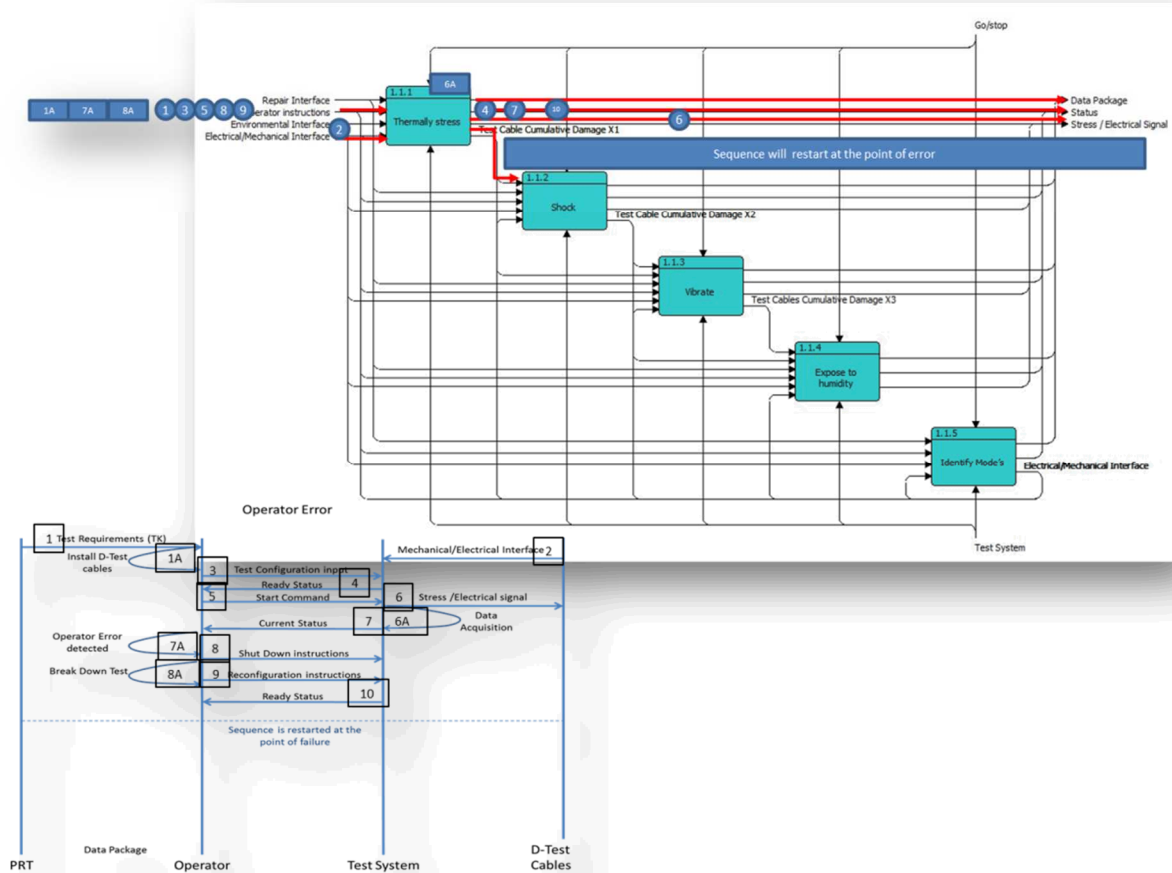


Figure 12 Operator Error Trace

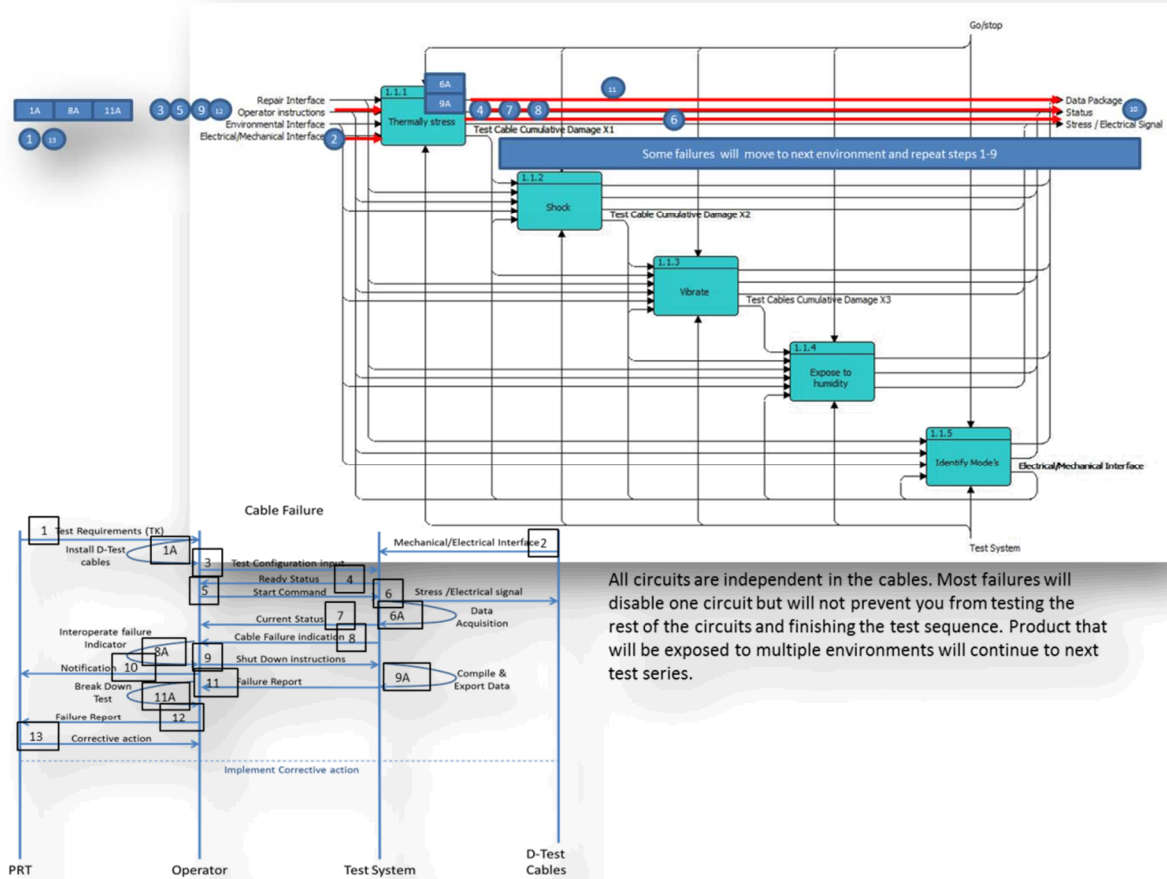


Figure 13 Cable Failure Trace

6. Create a physical architecture in CORE and relate it to the functional architecture you developed in #5.

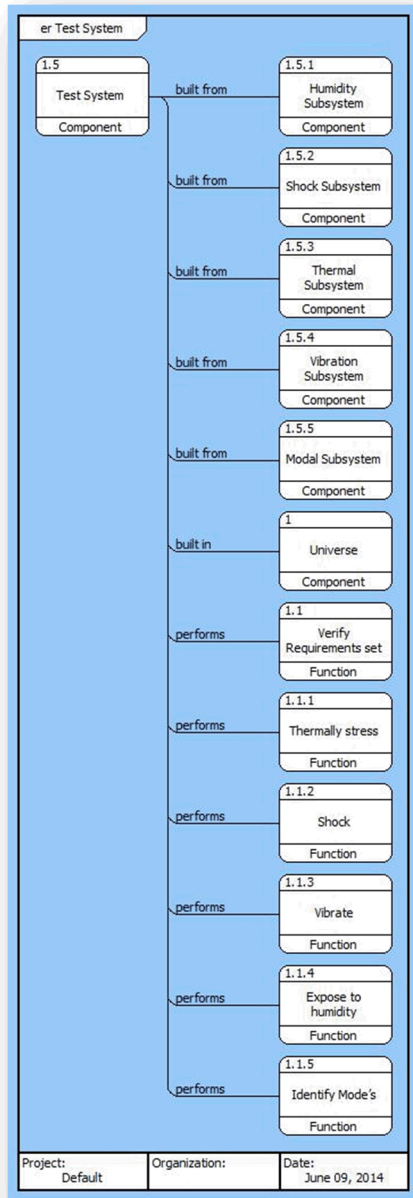


Figure 14

- a. Comment on the relationship between your functional and your physical architecture. Why did you allocate the specific functions to the specific components? Provide at least a half page discussion of the allocation rationale.

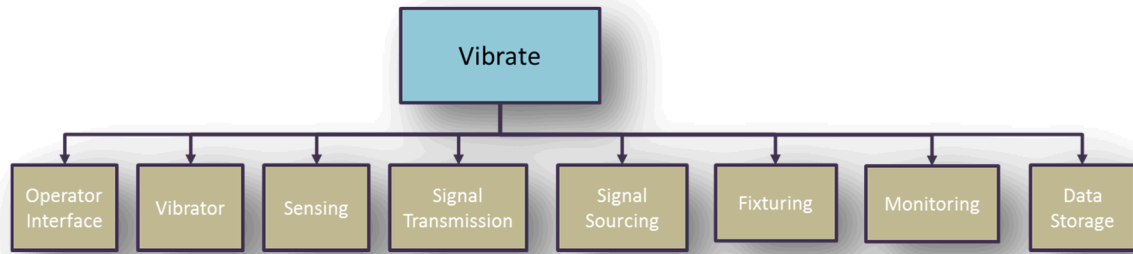
*It is at this point that the benefit of the selected architecture is realized. The selected architecture is based on the Physical Architecture. This results in a one to one relationship between components and the function. For example the humidity subsystem performs the expose to humidity function and the vibrate subsystem performs the vibrate function. In addition, any changes in components will not greatly impact the functionality of the overall system*

*based on this architecture, as long as the component is still able to perform as well or better than the previous component. Designing for simplicity adds value and is a desired attribute in this architecture. Using one component to perform more than one function is a possibility with this approach. One of the opportunities to perform two functions with one component is by using the Vibe subsystem to identify Modes. There is a method for gaining some insight to a components Modal response or resonant frequencies using a vibe table. Unfortunately the method of using the vibe table is not going to fully meet the customers expectation after a review of the requirements. There are more constraints and a more detailed characterization of the components modes, dampening, and mass properties required. As such, a separate subsystem is used to perform the Identify Modes function. I don't believe this architecture precludes implementing the discussed consolidation if cost or schedule drive a reduction in requirements. Unfortunately, the tradeoff for a simple architecture is a significantly more difficult mapping from requirements to the functional architecture. I am expecting the mapping between the requirements and the components or functions will be hard to follow. The mapping is not obvious in figure 14. The Humidity Subsystem performs the Expose to humidity function. The shock subsystem performs the shock function. The thermal subsystem performs the thermally stress function. The Vibration subsystem performs the vibrate function. Finally, the modal subsystem performs the identify modes function.*

7. Select the most complex first-level sub-function and repeat steps 4, 5 and 6 for this function.

Decompose the most complex first-level sub-function to create a first-level functional architecture. Provide a minimum of five functions in the first-level architecture. (Sep 4)

Document how else you could have partitioned the system function. What are the advantages and disadvantages of partitioning it the way you did? I am looking for alternative functional partitioning(s) that make sense, and thoughtful rationale of why you picked the one that you did as opposed to the alternative(s). Provide at least a one page discussion of your rationale. (Step 4a)



**Figure 15**

### Interface with operator

*I have considered three different methods of partitioning as I did at the first level partition. I considering Physical Architecture, Functional Requirements, and the Hatley-Pirbhai Template as the methods for defining the functional partitions. I have chosen a functional architecture based on the Physical Architecture. Figure 15 is the functional architecture that is based on the physical architecture. This architecture is flatter and not as deep, were the other architectures combine some of these functions or account for them at lower levels. There will not be as much one to one relation to the physical architecture but there is enough to justify the decision:*

- 1. The components must be exchangeable: As discussed above there will be a suite of equipment that the test house is permitted to use to conduct testing. This flexibility must be maintained at the subsystem level. If we were to overly restrict the test house, we would unnecessarily create a bottle neck that would be magnified with any competition for the specified resource. Considering these constraints it is critical to choose an architecture that allows component interchangeability.*
- 2. Our architecture must be developed with a limited catalog of components: This attribute must also be maintained at the sub-system level. The limited equipment that we can use and still meet our requirements introduces a level of conservatism. If I used another functional partition it would be more likely that I would develop an architecture that is not possible given our constraints.*
- 3. Legacy System: In truth this architecture represents a legacy architecture. At the moment implementing a innovative new approach adds too much risk given the schedule and cost constraints.*

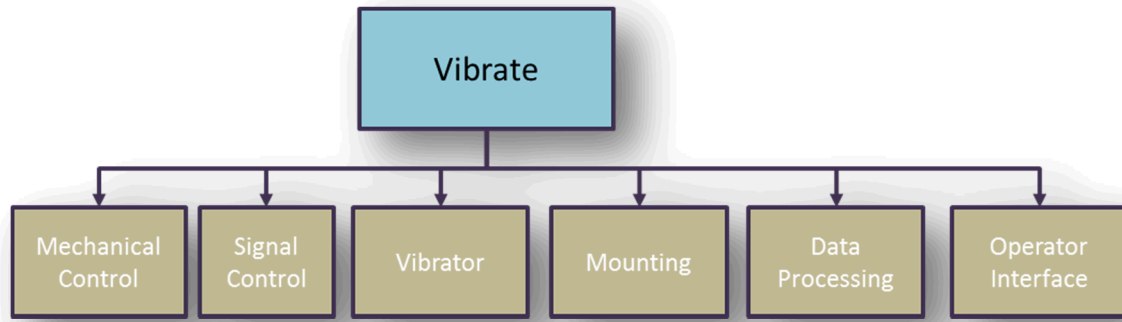


Figure 16 Hatley-Pirbhai sub-function

*If it was an option to develop a new system without the constraints discussed above, the Hatley-Pirbhai would have been the preferred approach. Although Figure 16 isn't a perfect fit with the Hatley-Pirbhai Template, this architecture is loosely based on it. The user interface would be accounted for at lower levels.*

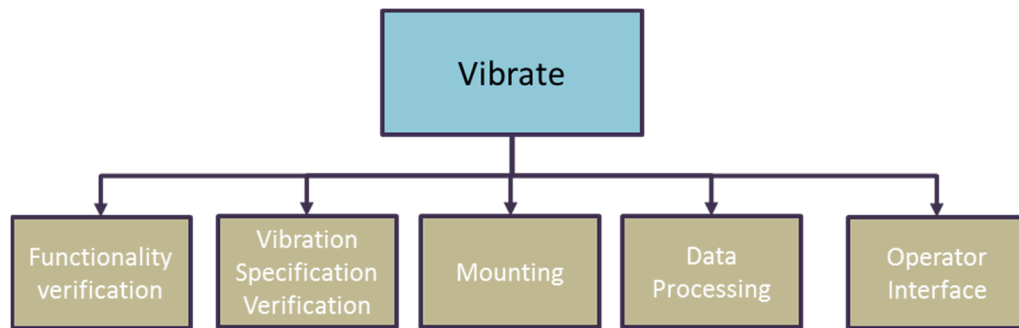
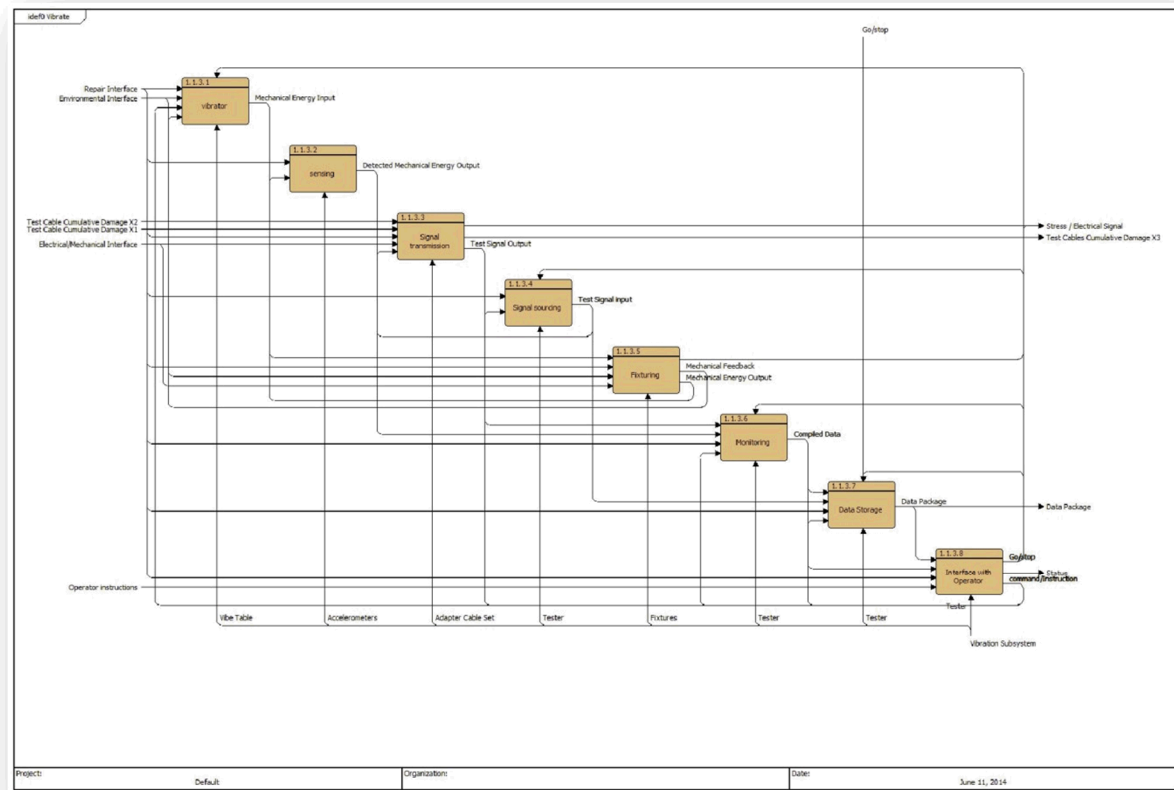


Figure 17 Functional Requirements sub-Function

*You can see that Figure 17 illustrates a functional requirements based architecture. Hence the Functionality verification and Vibration Specification Verification functions. These two functions are the two basic requirements of this subsystem.*

Develop an IDEF0 model in CORE for the first-level sub-function and the functional decomposition. Assign the external inputs and outputs to the system function and to the appropriate sub-functions. Create the necessary internal inputs and outputs to the first level architecture. (Step 5)



**Figure 18**

Document a trace of your four scenarios through your architecture. (Step 5a)

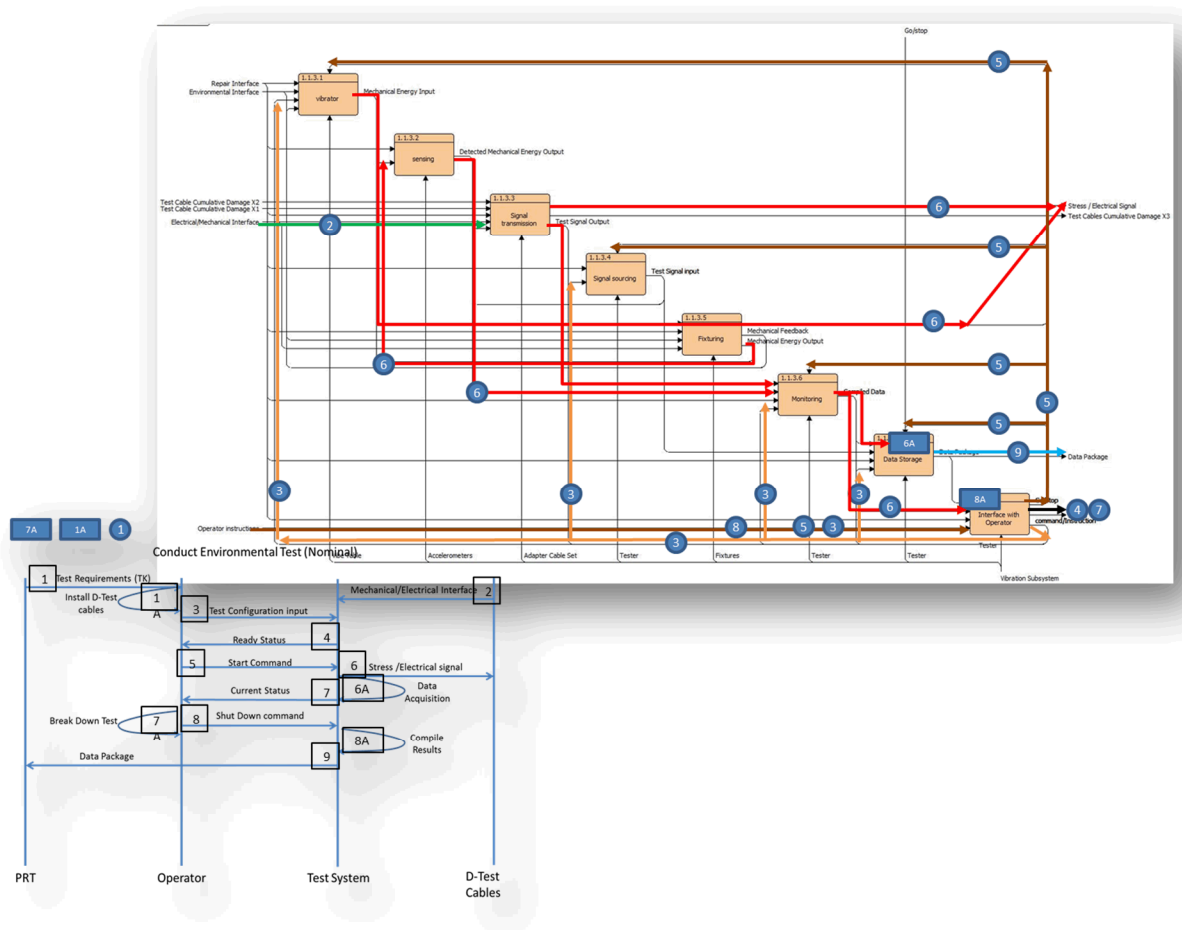


Figure 19 Sequence 2nd level Conduct Environmental Test (Nominal)

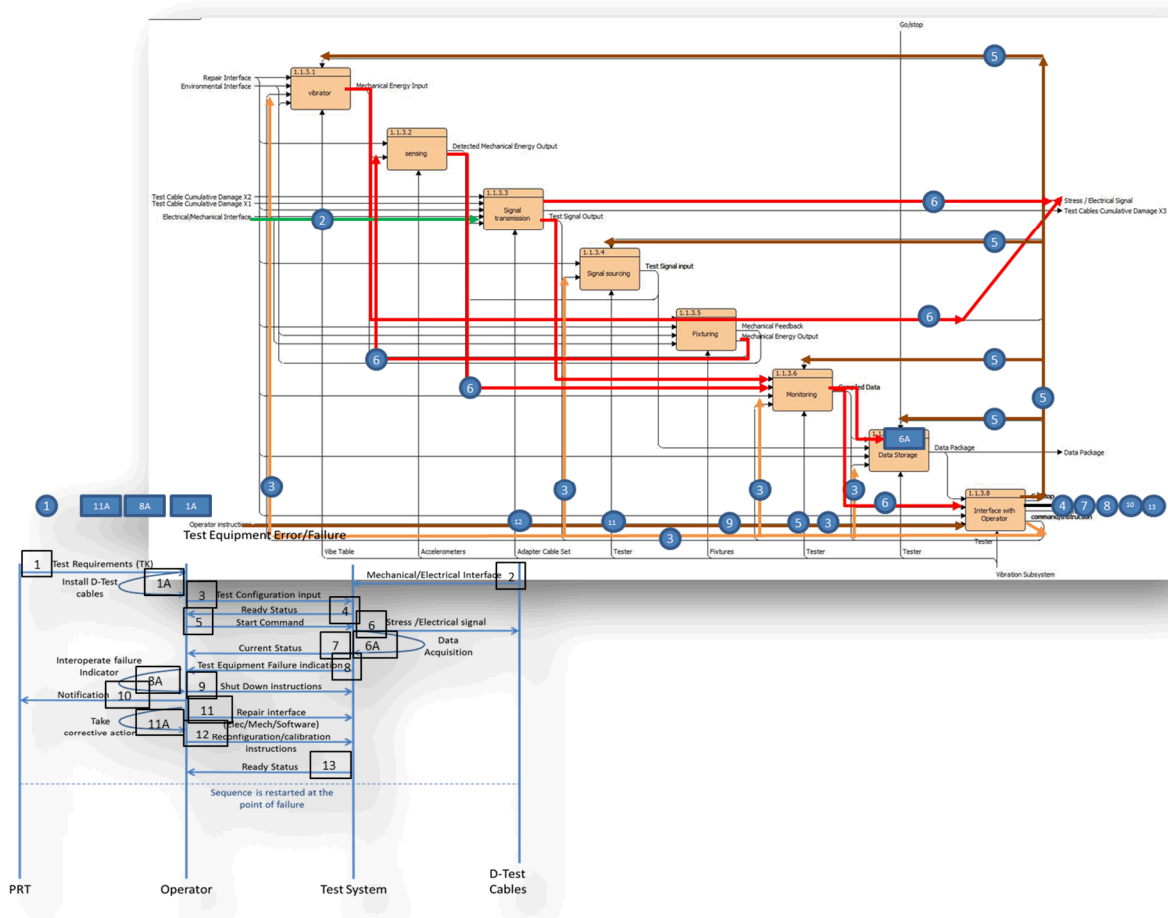


Figure 20 Sequence 2nd level Test Equipment Error/Failure

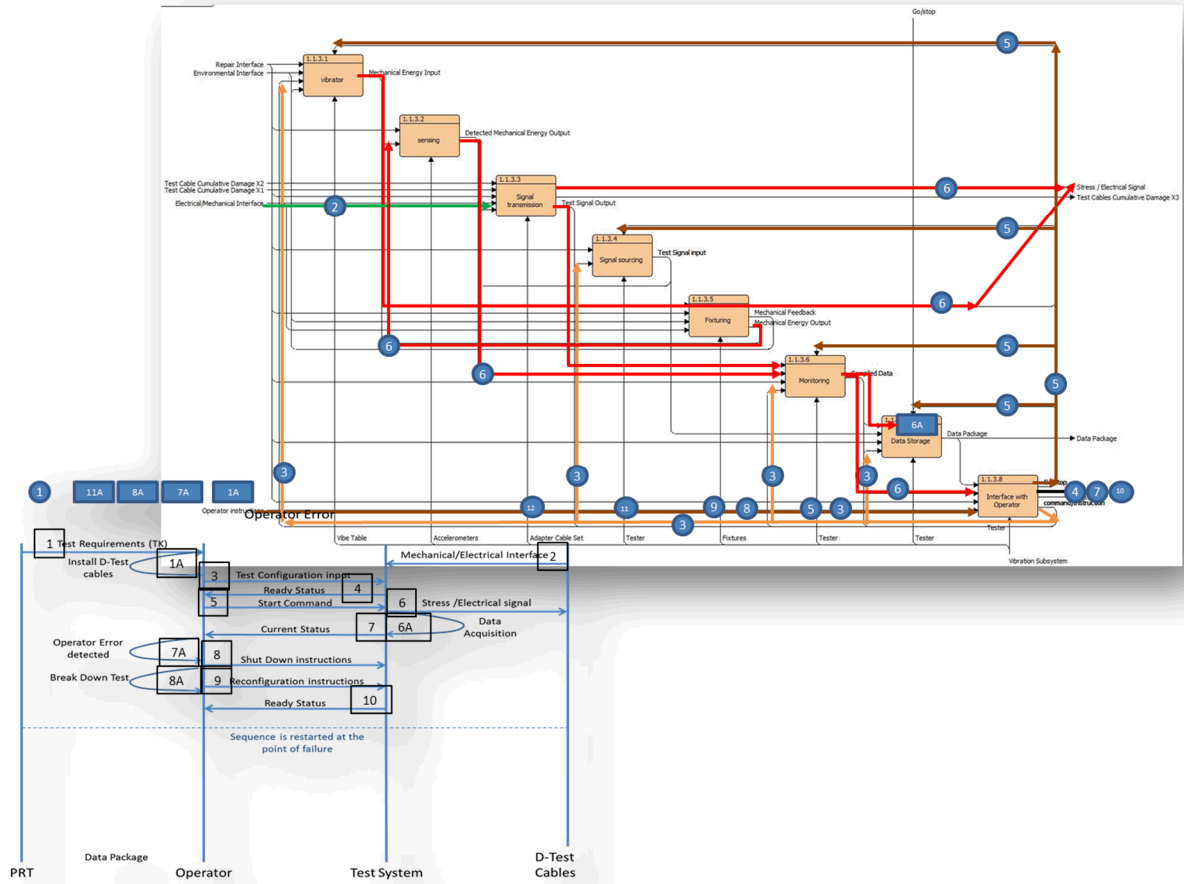
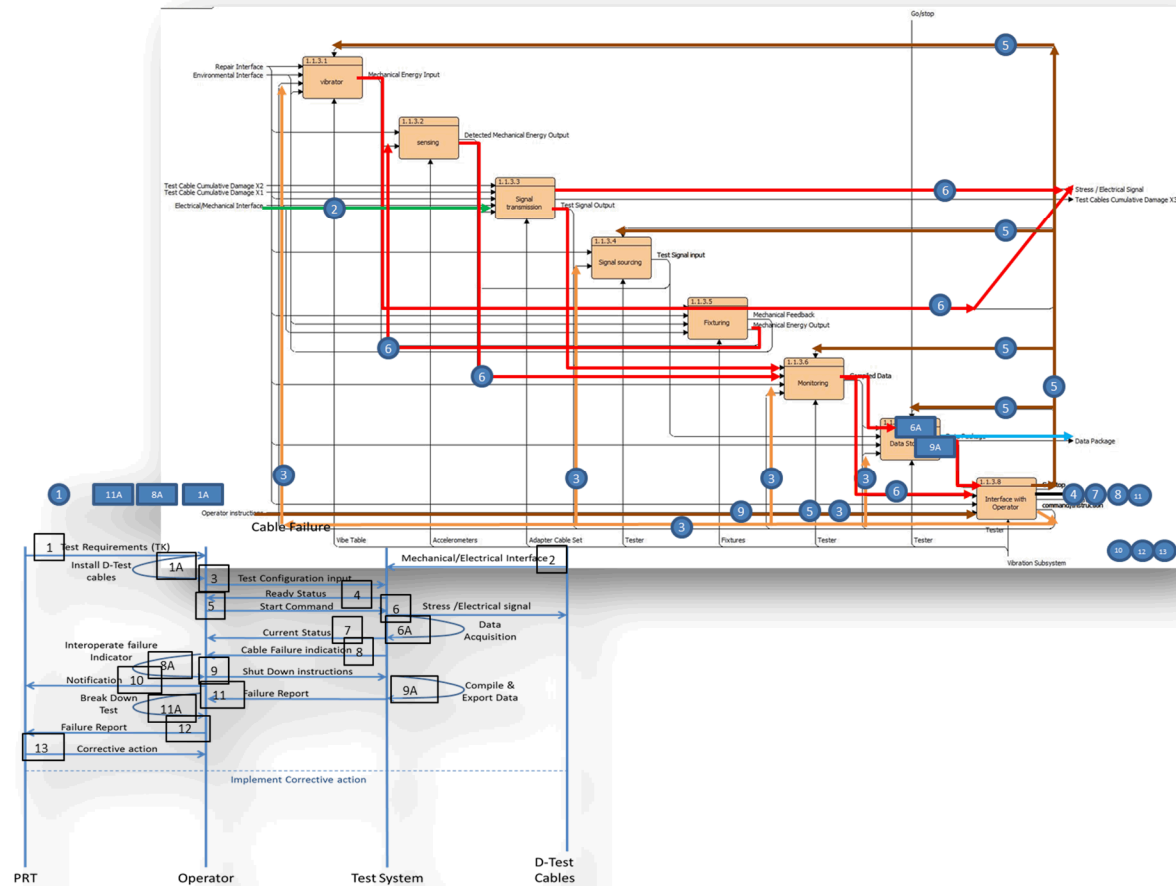


Figure 21 Sequence 2nd level Operator Error



**Figure 22 Sequence 2nd level Cable Failure**

Create a physical architecture in CORE and relate it to the functional architecture you developed for the sub-functions. (Step 6)

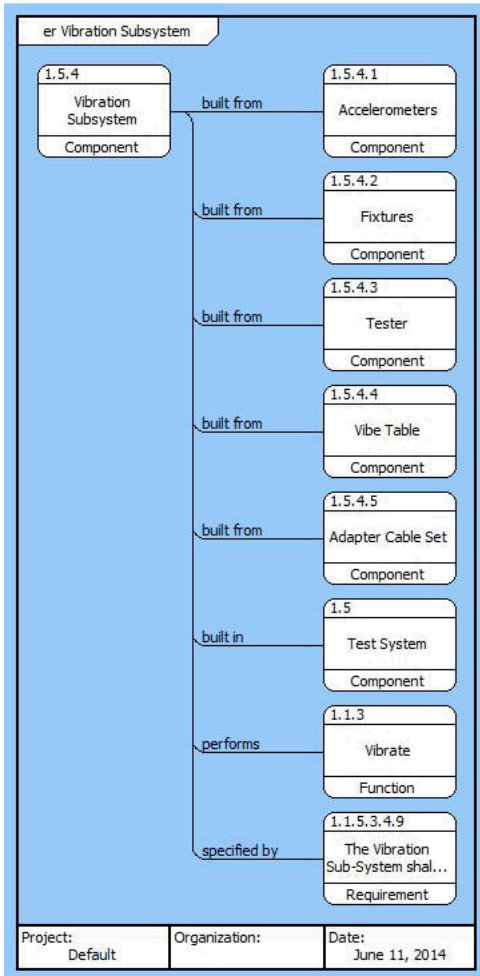


Figure 23 ER Vibration Subsystem

Comment on the relationship between your functional and your physical architecture. Why did you allocate the specific functions to the specific components? Provide at least a half page discussion of the allocation rationale. (Step 6a)

*This physical architecture is chosen based off of a legacy architecture. The objective of defining this system is to understand our system better, map our customer requirements to our current system and to identify any gaps. Any gaps will result in modification to the system based on the current constraints. The functional architecture was based on the legacy system. As such, the physical architecture is also based on the legacy system. The Accelerometer performs the sensing function. The Fixtures perform the fixturing function. The tester performs the Signal sourcing, Monitoring, data storage and interface with operator functions. The Vibe table performs that vibrator function. Finally the Adapter cable set performs the Signal Transmission function. The Tester is a system within a system. The tester is a very*

*complicated component of this physical architecture and performs three high level functions. The tester selection would be based on its ability to meet the requirements and perform the three basic functions as discussed. The breakdown of the tester would be the third level and would be invaluable to ensure the overall system meets its requirements.*

8. Create your physical connections (links in CORE) to connect the component that performs the first-level function you decomposed in #7 to the internal and external components involved in your four usage scenarios. Implement these links in CORE and relate the links to the items they transfer. If your architecture has more than 4 possible links, then provide 4 of them.

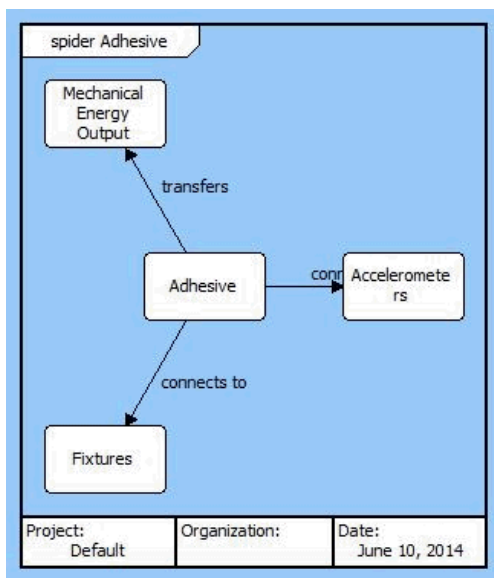


Figure 24 Spider Adhesive

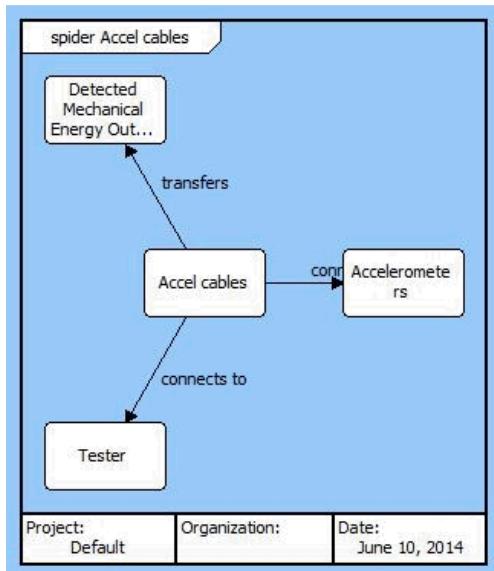


Figure 25 Spider Accel Cable

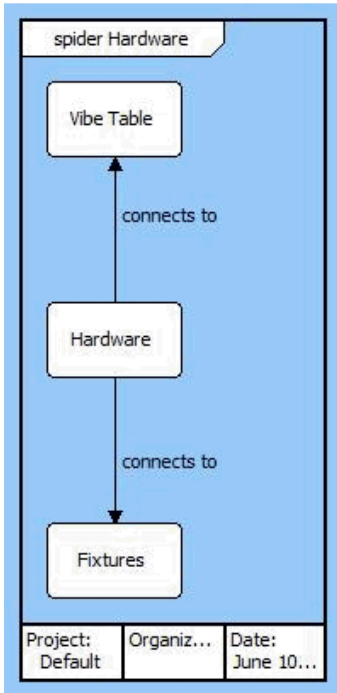


Figure 26 Spider Hardware

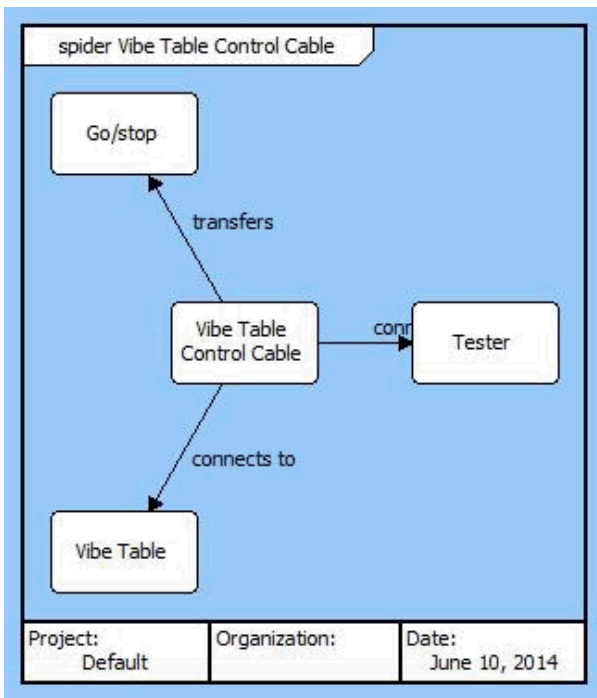


Figure 27 Spider Vibe Table Control Cable

9. Write a complete set of input, output, and functional requirements for the component whose function you decomposed in step #7. Enter these in CORE and relate (trace) them to the corresponding function in CORE.

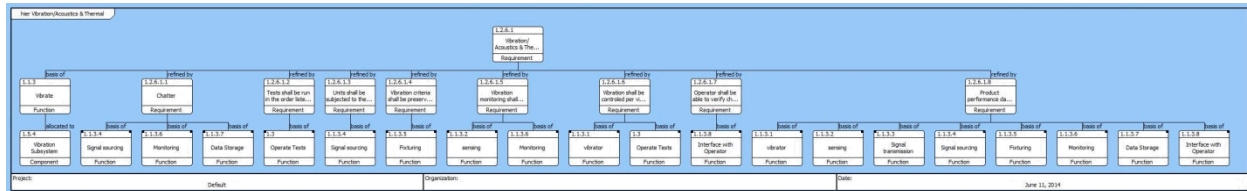


Figure 28 Requirements for Vibration Sub-System

10. Specify two system-level non-functional requirements that would be flowed down to the system's components for each of the following techniques: a) equivalence;



Figure 29 Equivalence flow down technique

- b) apportionment;

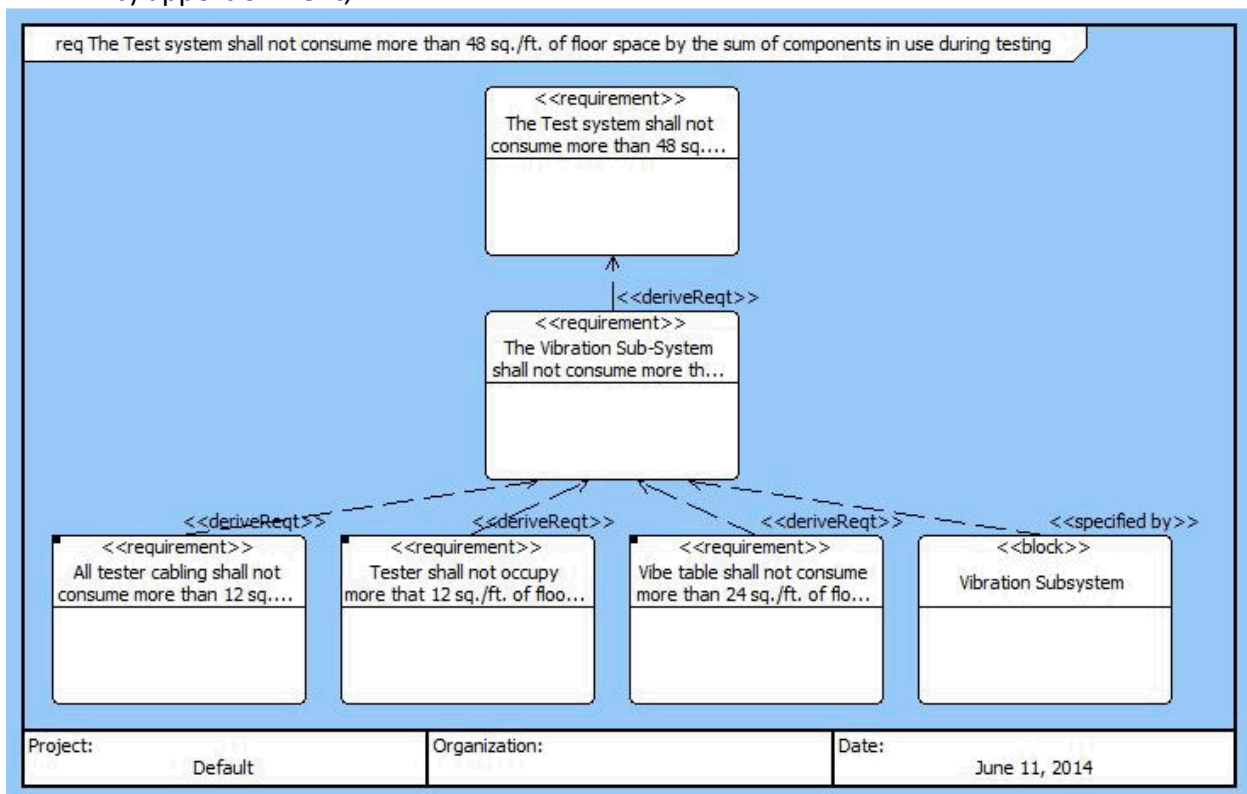


Figure 30 Apportionment flow down technique

- c) synthesis.

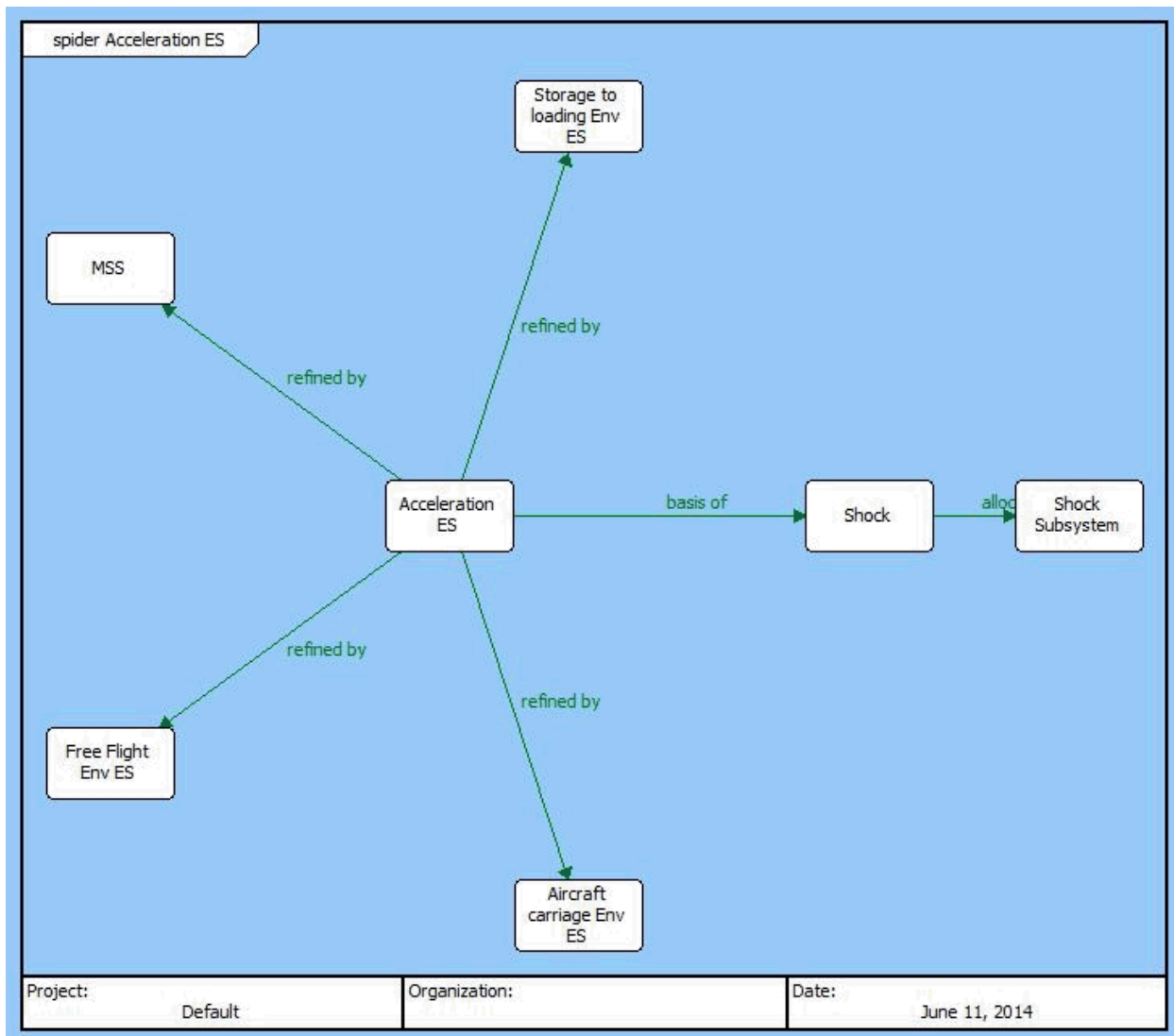


Figure 31 Synthesis flow down technique

What might be the corresponding requirements for the component whose function you decomposed in step #7? Enter both the system and corresponding component requirements in CORE, provide the traceability between the pairs of requirements in CORE (parent/child relationship), and trace all of the requirements to the appropriate function or component. Each student will define a total of 12 requirements (6 system-level and 6 associated component-level).

*Equivalence is a simple flowdown technique that causes the subsystem requirement to be the same as the system requirement. The Equivalence requirement flow is the CD Functional Requirement stating that the Cables functionality shall be verified during and after every environmental test. The verification of the functionality of the cable is a requirement for the overall system and must be verified at all levels of*

*environments. The apportionment requirement states that “The Test system shall not consume more than 48 sq./ft. of floor space by the sum of components in use during testing”. This kind of apportionment requirement is common for production facilities that have to manage floor space. The Synthesis requirement is the Acceleration ES requirement that states “Product shall function through acceleration environment defined in ES”. This is considered Synthesis because it will be verified by analysis with the combination of Modal data and the products performance in shock environments. Since shock and acceleration is defined in G’s, you can analyse the products susceptibility to acceleration damage if you know where the products resonant frequencies reside in the frequency spectrum. the Diagrams in figures 29-31 were selected to best illustrate the flowdown techniques.*

11. Create a Subsystem (Component) Description Document (SDD) for the component that performs the function that you decomposed to the second level. (Submit the SDD as part of your homework.)

*Please see SDD file “SDOE 650 Colin George SDD”*

12. For your system: a) Identify several criteria you think would be important for assessing your system architecture and describe why they are important, b) Select one criterion for which you think the architecture you presented would score well. Why do you think so? c) Select an important criterion that you think would be a challenge for your architecture. Why would that be so? What might you do about it? Provide at least a one page discussion.

*a) The criteria that would best assess this system architecture is to answer the following question. “Does it meet its overall requirement to verify the basic system level requirement?” The basic system level requirement was to verify all cable requirements that are planned to be verified by Test. Additional criteria is that the system fits within its allotted budget, meets schedule for Data needs, supports the reliability requirement, and supports product qualification.*

*This test system was not meant to verify all cable requirements. To verify an entire set of requirements by test indiscriminately without evaluating other methods of verification is cost and schedule prohibitive. The high cost associated with test is why*

*modeling and simulation has become such an asset in engineering. The data from this test system will be used to validate our cable models. Besides this many requirements are simply not testable. Many of the cables requirements will be verified by demonstration, inspection, and by analysis. By defining the Test Systems physical architecture, functional architecture and mapping them to the requirements architecture, I can greatly increase confidence in the cables ability to meet system requirements.*

*The criteria to fit within an allocated budget will often be found associated with any effort. This System is no different. The systems cost will primarily be a rental cost since a vendor will procure, maintain, man and operate this system. It is expected that all of the contractors costs and for profit fee will summed into the customers expense. With this in mind the requirements for the system and the systems architecture must not excessively drive up vendors costs.*

*In some cases the program schedule is more critical than cost. Even though cost and schedule are related, when a unit of time carries more impact to a program than the associated unit of cost the schedule becomes more critical. In most programs qualification and development cannot continue until data has validated current design decisions. When that data is not delivered, the program must decide to move ahead at risk or stop development until the activity is complete. Both moving ahead at risk and stopping is not acceptable from a schedule perspective. For this reason the test architecture must be designed in such a way that procurement and execution fits within the allotted schedule time.*

*The overall objective of the Cables requirements is to meet its reliability requirement. The reliability invokes functionality within the environment. Confidence invokes statistical significances. If the product functions properly within the specified environments or fails well above the specified environments the data will verify the reliability requirement. If enough cables are tested and variable data is collected confidence will be demonstrated. These work together to demonstrate or prove*

*reliability.*

*Finally, another major objective of the cable realization effort is to qualify the cables design. This qualification requires evidence that proves that the cables meet requirements including reliability. In addition, qualification characterizes the product, allowing us to understand its failure modes, susceptibility, and robustness. If for whatever reason the requirements for the system change in the future, characterization will mitigate the risk of forcing a requalification effort. A good qualification can save lots of money later in the product's life cycle.*

*b) I expect that this architecture will score well in verifying all requirements that must be tested. The reason for this is because its purpose was to do just that. I created the cables requirements architecture before I did anything else. The cables requirements set could not be decreased or loosened. During the physical and function architecture development, this requirement set aided in every decision. Even though the functional architecture was based on the physical architecture and the physical architecture was based on a legacy design, this architecture integrates a suite of components to meet requirements. Even though requirements were set and new components designs were not possible, there was enough freedom in component selection to allow an integrated system that meets requirements.*

*c) The architecture's ability to support the reliability requirement will be the most challenging for this architecture. Even though I designed in the capability to run Cable product through the system in a variety of sequences, the statistical basis will be hard to defend given the Sandian culture. There are too many subject matter experts that can pick apart the defined approach. The best way to meet this criteria is to socialize this architecture with the customer and a hand full of the most respected Subject matter experts at Sandia. After socializing the architecture and making changes according to feedback we could acquire a written letter from the customer. These two steps will increase the likelihood that we meet this criteria.*

13. The above process is not viewed as a sequential and linear process, but rather as an

iterative and recursive set of activities and decisions, with subsequent decisions providing more insight into the earlier steps in the synthesis process. Document this iterative process as it played out during your project definition (provide specific examples). Also, comment on the value of a tool like CORE for managing the architectural data and relationships. Specifically, document what in CORE was helpful or detrimental for developing and understanding your project's architecture. Provide at least a one page discussion.

*When I read this question I couldn't help but smile in light of the evolution of this architecture. Steps that I expected to be trivial, turned out to be more difficult and drove more changes than I expected. Many times during the development of this architecture I made many changes retroactively. The retroactive changes consumed allot of time. Changes happened at almost every point of the architecture development and at all levels of the system.*

*At the context diagram level or the meta-function level, I originally had requirements as an output from the PRT to the Test system. I also had status as an output of the system to the PRT. After moving on to the sequence diagrams I could see that this didn't make any sense. I knew that we would not receive status updates directly and that the system would not receive requirements directly. I made those as inputs and outputs between the operator and the PRT but I established the data package as an interface between the Test system and the PRT. I could have removed the PRT as an active stakeholder by routing the data package and report through the operator but I wanted to preserve the PRT in the sequence diagram and ensure that we get our data from the source/Test system. I don't want the data interpreted for us or altered in anyway.*

*At the test system level I started out with functions such as "Simulate Margin" and "Simulate Aging" which were removed latter. The reason I removed them was because Simulating Aging was redundant with the thermal function and Simulating Margin was wrong. Margin is demonstrated and verified by analysis using the test data. The items would often need to be remapped or changed after changes at the subsystem level.*

*The activity that produced the most changes was the tracing of my four scenarios at the system level and subsystem level. This effort was very time consuming primarily because of the changes that were required. I expected the sub-system level trace to be*

*trivial but it turned out to be the most complicated. Upon tracing the second step of my sequence I saw that I did not create an item to connect the system to the D-test cable. The changes that occurred to my initial sub-system functional architecture was substantial. I originally forgot to include an operator interface function as well as data storage. As I traced the sequence I would often find that the trace ended at a function and didnt continue as it should have. I also had inputs that had to be changed because they didn't make sense at the subsystem level or I had to ad functions to handle inputs and create the outputs. The presentation of the resulting architecture may seem obvious to an individual who knows test systems by the creation of it was not obvious and not smooth. Over the course of completing this assignment I have been seriously contemplating the purchase of this software. Its graphical representation of the architecture is so much cleaner than doing it in Visio or PowerPoint. The multiple dimensions that this software models is very valuable. Creating a physical architecture is good, creating a functional architecture is also good, but being able to create a requirements architecture and map it to the physical and function architecture is invaluable. As a comparison DOORs maps at one dimension in a table format, core maps multiple dimensions and provides them in a diagram. Both table formats and diagram formats have value but having both allows you to maximize value according to your situation and application. I have not yet looked for a price for one license. I am hoping I will be able to purchase it for work.*